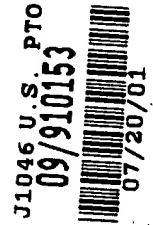


日 本 国 特 許 庁
JAPAN PATENT OFFICE



別紙添付の書類に記載されている事項は下記の出願書類に記載されて
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office

出 願 年 月 日

Date of Application:

2000年12月 7日

出 願 番 号

Application Number:

特願2000-372578

出 願 人

Applicant(s):

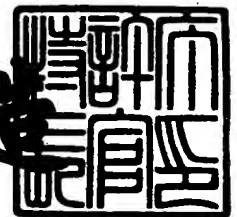
株式会社日立製作所

CERTIFIED COPY OF
PRIORITY DOCUMENT

2001年 5月11日

特許庁長官
Commissioner,
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3040108

【書類名】 特許願

【整理番号】 NT00P0815

【提出日】 平成12年12月 7日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 3/06

【発明者】

 【住所又は居所】 神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所 システム開発研究所内

 【氏名】 水野 陽一

【発明者】

 【住所又は居所】 神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所 システム開発研究所内

 【氏名】 松並 直人

【発明者】

 【住所又は居所】 神奈川県小田原市国府津 2 8 8 0 番地 株式会社日立製作所 ストレージシステム事業部内

 【氏名】 村岡 健司

【発明者】

 【住所又は居所】 神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所 システム開発研究所内

 【氏名】 味松 康行

【特許出願人】

 【識別番号】 000005108

 【氏名又は名称】 株式会社日立製作所

【代理人】

 【識別番号】 100068504

 【弁理士】

 【氏名又は名称】 小川 勝男

 【電話番号】 03-3661-0071

【選任した代理人】

【識別番号】 100086656

【弁理士】

【氏名又は名称】 田中 恭助

【電話番号】 03-3661-0071

【選任した代理人】

【識別番号】 100094352

【弁理士】

【氏名又は名称】 佐々木 孝

【電話番号】 03-3661-0071

【手数料の表示】

【予納台帳番号】 081423

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 計算機システム

【特許請求の範囲】

【請求項 1】

少なくとも 1 台以上の計算機と、該計算機が使用できる論理ボリュームを持つストレージと、該ストレージの論理ボリューム構成の変更を指示する管理ユーティリティを備えた計算機システムにおいて、

前記計算機は前記ストレージの論理ボリュームの構成変更を認識する論理ボリューム認識手段を備え、

前記ストレージは論理ボリュームの構成制御を行なう論理ボリューム制御手段を備えるとともに、該論理ボリューム制御手段は、論理ボリュームの構成情報を記述する論理ボリューム番号マップを備え、

前記論理ボリューム制御手段は、前記管理ユーティリティからの指示に従い、前記論理ボリューム番号マップに記述されている前記計算機から見える論理ボリューム番号とこの計算機から見える内部論理ボリューム番号に対応する前記ストレージの内部論理ボリューム番号の割り合てを変更し、前記計算機から見える論理ボリュームに対応する前記ストレージの内部論理ボリューム番号の割り合てを複数とすることを可能とすることを特徴とする計算機システム。

【請求項 2】

前記論理ボリューム番号マップは、前記計算機から見える論理ボリューム番号と、前記ストレージ内部の論理ボリューム番号と、計算機から見える論理ボリューム番号と前記ストレージの内部論理ボリューム番号との結合関係を記述する論理ボリューム番号結合情報とを備え、前記論理ボリューム制御手段は、前記管理ユーティリティからの指示に従い、計算機との対応が未定義のストレージの内部論理ボリュームに対して計算機から見える論理ボリューム番号と前記ストレージの内部論理ボリューム番号との新しい結合関係を論理ボリューム番号マップに記述して論理ボリュームの構成変更を行ない、前記論理ボリューム認識手段によって前記計算機に割り当てられた論理ボリュームの構成変更が認識される請求項 1 記載の計算機システム。

【請求項 3】

前記論理ボリューム番号マップには、前記内部論理ボリュームの大きさを示すブロック数が記述されるとともに、複数の内部論理ボリュームが割り当てられたときはそれぞれの内部論理ボリュームの先頭アドレスを直前までの内部論理ボリュームのブロック数の累積とする請求項 1 または 2 記載の計算機システム。

【請求項 4】

少なくとも 1 台以上の計算機と、該計算機が使用できる論理ボリュームを持つストレージと、該ストレージの論理ボリューム構成の変更を指示する管理ユーティリティを備えた計算機システムにおいて、

前記計算機は前記ストレージの論理ボリュームの構成変更を認識する論理ボリューム認識手段を備え、

前記ストレージは論理ボリュームの構成制御を行なう論理ボリューム制御手段を備えるとともに、該論理ボリューム制御手段は、論理ボリュームの構成情報を記述する論理ボリューム番号マップおよび前記論理ボリュームのコピーを行なうコピー手段を備え、

前記論理ボリューム制御手段は、前記管理ユーティリティからの指示に従い、前記論理ボリューム番号マップの記述から未定義の内部論理ボリューム領域の容量と前記管理ユーティリティの要求する容量との比較を行い、未定義の内部論理ボリューム領域の容量が大きい場合にのみ、前記未定義の内部論理ボリューム領域に新たな内部論理ボリュームを定義して前記管理ユーティリティの指定する論理ボリューム番号の領域の記録内容をコピーするとともに、コピー開始を前記計算機に通知し、コピー完了に応じて新たに定義された内部論理ボリューム領域に新たな内部論理ボリューム番号を設定することを特徴とする計算機システム。

【請求項 5】

前記論理ボリューム番号マップは、新たに定義された内部論理ボリューム領域に新たな内部論理ボリューム番号が設定されたとき、前記管理ユーティリティの指定する論理ボリューム番号に対応する領域を、前記計算機との対応が未定義の内部論理ボリュームに変更する請求項 4 記載の計算機システム。

【請求項 6】

前記計算機は、前記コピー手段によるコピーが行われることを通知されたときは前記コピー手段によるコピーが行なわれている間、前記計算機から前記ストレージへのアクセスは、前記管理ユーティリティの指定する論理ボリューム番号に対応する領域と新たに定義されるべき内部論理ボリューム領域とに並行して行うものである請求項4記載の計算機システム。

【請求項7】

少なくとも1台以上の計算機と、該計算機が使用できる論理ボリュームを持つストレージと、該ストレージの論理ボリューム構成の変更を指示する管理ユーティリティを備えた計算機システムにおいて、

前記計算機は前記ストレージの論理ボリュームの構成変更を認識する論理ボリューム認識手段を備え、

前記ストレージは論理ボリュームの構成制御を行なう論理ボリューム制御手段を備えるとともに、該論理ボリューム制御手段は、論理ボリュームの構成情報を記述する論理ボリューム番号マップおよび前記論理ボリュームのコピーを行なうコピー手段を備え、

前記論理ボリューム制御手段は、前記管理ユーティリティからの指示に従い、前記論理ボリューム番号マップの記述から未定義の内部論理ボリューム領域の容量と前記管理ユーティリティの要求する容量との比較を行い、未定義の内部論理ボリューム領域の容量が大きい場合は、前記未定義の内部論理ボリューム領域に新たな内部論理ボリュームを定義して前記管理ユーティリティの指定する論理ボリューム番号の領域の記録内容をコピーするとともに、コピー開始を前記計算機に通知し、コピー完了に応じて新たに定義された内部論理ボリューム領域に新たな内部論理ボリューム番号を設定し、前記未定義の内部論理ボリューム領域の容量が不足の場合には、前記管理ユーティリティからの指示に従い、前記論理ボリューム番号マップに記述されている前記計算機から見える論理ボリューム番号とこの計算機から見える内部論理ボリューム番号に対応する前記ストレージの内部論理ボリューム番号の割り合てを変更し、前記計算機から見える論理ボリュームに対応する前記ストレージの内部論理ボリューム番号の割り合てを複数とすることを可能とすることを特徴とする計算機システム。

【請求項 8】

前記論理ボリューム番号マップは、前記計算機から見える論理ボリューム番号と、前記ストレージ内部の論理ボリューム番号と、計算機から見える論理ボリューム番号と前記ストレージの内部論理ボリューム番号との結合関係を記述する論理ボリューム番号結合情報とを備え、前記論理ボリューム制御手段は、前記管理ユーティリティからの指示に従い、計算機との対応が未定義のストレージの内部論理ボリュームに対して計算機から見える論理ボリューム番号と前記ストレージの内部論理ボリューム番号との新しい結合関係を論理ボリューム番号マップに記述して論理ボリュームの構成変更を行ない、前記論理ボリューム認識手段によって前記計算機に割り当てられた論理ボリュームの構成変更が認識される請求項 7 記載の計算機システム。

【請求項 9】

前記論理ボリューム番号マップには、前記内部論理ボリュームの大きさを示すブロック数が記述されるとともに、複数の内部論理ボリュームが割り当てられたときはそれぞれの内部論理ボリュームの先頭アドレスを直前までの内部論理ボリュームのブロック数の累積とする請求項 7 または 8 記載の計算機システム。

【請求項 10】

前記論理ボリューム番号マップは、新たに定義された内部論理ボリューム領域に新たな内部論理ボリューム番号が設定されたとき、前記管理ユーティリティの指定する論理ボリューム番号に対応する領域を、前記計算機との対応が未定義の内部論理ボリュームに変更する請求項 7 記載の計算機システム。

【請求項 11】

前記計算機は、前記コピー手段によるコピーが行われることを通知されたときは前記コピー手段によるコピーが行なわれている間、前記計算機から前記ストレージへのアクセスは、前記管理ユーティリティの指定する論理ボリューム番号に対応する領域と新たに定義されるべき内部論理ボリューム領域とに並行して行うものである請求項 7 記載の計算機システム。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、ストレージ装置内における論理ボリューム領域の構成変更の方法に関し、特に論理ボリューム領域拡張の方法に関する。

【 0 0 0 2 】

【従来の技術】

従来、ストレージ装置内の容量を拡張する方法として、物理ディスク追加による拡張方法が知られている。図 1 6 を参照して説明する。1 0 0 はストレージ装置、1 0 1 は物理ディスクの集合からなる物理ディスク領域である。物理ディスク領域 1 0 1 には、LU (Logical Unit) と呼ばれる論理ボリュームが構成される。1 0 1 1 x (1 0 1 1 a, 1 0 1 1 b) は論理ボリュームである。ストレージの容量を拡張したい場合、ストレージ 1 0 0 に物理ディスクを追加することによって、破線で示すように、追加ディスク領域 1 0 2 が生まれる。物理ディスク領域 1 0 1 の論理ボリューム LU 1 (1 0 1 1 a)、LU 2 (1 0 1 1 b) が一杯になったときは、追加ディスク領域 1 0 2 に新たに論理ボリューム LU 3 (1 0 1 1 c) を作成することができる。

【 0 0 0 3 】

しかし、物理ディスクの追加によらず、すでに配分がなされている内部の論理ボリューム領域の割り当てを拡張して使用可能なストレージの容量を拡張しようとする問題が発生する。例えば、図 1 6 で LU 1 (1 0 1 1 a) の領域を拡張しようとする、すぐ後ろに連続的に LU 2 (1 0 1 1 b) の領域があるため、LU 1 (1 0 1 1 a) の領域を拡張するためには LU 2 (1 0 1 1 b) の領域を、拡張する領域に対応した分だけ、解放しなくてはならない。また、ストレージ装置 1 0 0 内の論理ボリューム領域を拡張するには、一旦その領域を解放し、新たに論理ボリューム領域の配分を定義し直す必要があるため、論理ボリュームへのアクセスを中断しなければならない。

【 0 0 0 4 】

【発明が解決しようとする課題】

このように、システムの運用中であっても、物理ディスクを増設して論理ボリ

ューム領域の容量を拡張することは簡単である。一方、論理ボリュームは内部の連続領域に構成されるため、物理ディスクを増設することなく、システムの運用中に論理ボリューム個別の領域を拡張することはできない。すなわち、ストレージ装置内の論理ボリューム領域を拡張するには、一旦その領域を解放し、新たに論理ボリューム領域を定義し直す必要があるため、システムの運用中は論理ボリュームの領域の変更はできないのである。

【 0 0 0 5 】

本発明の目的は、ストレージ装置内部の論理ボリューム領域の拡張等の論理ボリュームの構成を、システムの運用中に、自由に変更することを可能にすることである。すなわち、既存の論理ボリューム領域を解放せずに論理ボリューム領域を拡張することを可能にし、既存の論理ボリュームへのアクセスを中断させることなく論理ボリューム領域を拡張する方法を提供することである。

【 0 0 0 6 】

【課題を解決するための手段】

本発明が適用されて効果を得られる計算機システムは、たとえば、1台以上の計算機と、ストレージと、ストレージの論理ボリューム構成の変更指示を行なう管理ユーティリティとから構築される。

【 0 0 0 7 】

本発明は、ストレージは論理ボリュームの構成制御を行なう論理ボリューム制御手段を備えるとともに、論理ボリューム制御手段に、論理ボリュームの構成情報を記述する論理ボリューム番号マップを備える。この論理ボリューム番号マップに計算機と論理ボリュームの構成を記述し、計算機が使用可能なストレージ内部の論理ボリュームとの組み合わせを記述することによって論理ボリュームの自由な構成変更を可能とするものである。

【 0 0 0 8 】

【発明の実施の形態】

以下、本発明を実施例を示す図を参照して説明する。

【 0 0 0 9 】

(実施例 1)

図 1 は本実施例 1 の計算機システムの構成図である。図において、2 x (2 a, 2 b, ---, 2 n) は計算機、1 はすべての計算機 2 x が共用するストレージ、4 は計算機システムを管理するための管理コンソール、3 はすべての計算機 2 x とストレージ 1 と管理コンソール 4 を相互に接続するファイバチャネル接続手段、5 x (5 a, 5 b, ---, 5 n) はファイバチャネルである。6 x (6 a, 6 b, ---, 6 n) は複数の計算機 2 x と管理コンソール 4 が通信するための LAN (Local Area Network) である。7 はストレージ 1 と管理コンソール 4 が通信するための通信手段である。

【 0 0 1 0 】

各計算機 2 x において、2 1 x (2 1 a, 2 1 b, ---, 2 1 n) は論理ボリューム管理ソフトであり、ファイルシステム F S 又は一般に L V M と呼ばれる。2 1 1 x (2 1 1 a, 2 1 1 b, ---, 2 1 1 n) は論理ボリューム認識手段であり、ストレージ 1 の論理ボリューム構成等を認識し、構成の変更を論理ボリューム管理ソフト 2 1 x に通知する。ここで、論理ボリュームとは、ストレージ 1 内部に設けた仮想的なボリュームのことであり、計算機 2 x とストレージ 1 とを接続するインターフェイスの一つのプロトコルである S C S I (Small Computer System Interface) の仕様において定義された名称である。以下、論理ボリュームのことを単に L U (Logical Unit) と呼ぶことがある。また L U を識別するための番号のことを論理ボリューム番号 (L U N : Logical Unit Number) と呼ぶ。

【 0 0 1 1 】

管理コンソール 4 において、4 1 は管理ユーティリティであり、ストレージ 1 内部の L U 構成の表示や、システムの管理者がストレージ 1 の L U 設定等を行なう為に使用される。ここでは管理ユーティリティ 4 1 は管理コンソール 4 上に配置するものとしたが、計算機 2 x や、ストレージ 1 に配置してもかまわない。

【 0 0 1 2 】

ストレージ 1 において、1 1 は論理ボリューム制御手段であり、ストレージ 1 内の論理ボリューム構成を制御し、各計算機 2 x の論理ボリューム認識手段 2 1 1 x からの要求に応じて論理ボリューム構成情報を通知する。1 1 1 は論理ボリ

ューム番号マップ（LUNマップ）であり、LU制御手段11の構成要素として、ストレージ1の論理ボリュームと計算機2xから認識できる論理ボリュームとの論理的な対応を示すマップである。12は物理ディスク領域であり、物理ディスクの集合からなる。物理ディスク領域12には、論理的に作成された論理ボリューム121x（121a, 121b, ---, 121n）が構成される。

【0013】

論理ボリュームLUについて説明する。LUはストレージ1の上位計算機2x（2a, 2b, ---, 2n）から見たときの論理的なボリュームである。上位計算機は1つのLUを1台の論理的なディスク装置として認識する。ストレージ1は内部に複数のLUを定義し、構築することができる。これを内部論理ボリューム（内部LU）と呼ぶことにする。ストレージ1では内部LUを管理するため0から始まる整数でシリアル番号付けする。この番号を内部論理ボリューム番号（内部LUN）と呼ぶ。

【0014】

一般に計算機はOSブート時に接続するストレージをサーチしてLUを検出するが、サーチ方法にいくつかの制約がある場合がある。それは、

（a）論理ボリューム番号LUNは0から順にサーチする。

（b）論理ボリューム番号LUNは連続番号で存在することを仮定し、ある番号が存在しない場合それ以降のサーチは行わない。

の2点である。これはサーチ時間を短縮するための工夫である。

【0015】

本発明の計算機もこのような特性をもつ計算機2xであると仮定する。このような場合、内部論理ボリューム番号LUNをそのまま上位計算機に割り当てるとすると、内部LUN=0以外の番号を割り当てられた計算機はこのLUを検出できないことになってしまう。すなわち、どの計算機も、自分が使う論理ボリューム番号LUNは0から始まることを前提としているから、内部論理ボリューム番号が直接論理ボリューム番号LUNとして割り当てられたのでは内部LUN=0以外の番号を割り当てられた計算機から見ると論理ボリュームLUが割り当てられていないことと同じになってしまう。そこで、すべての計算機に対し、計算機

から見たときの論理ボリューム番号が 0 から始まり、かつ連続番号で LUN を割り当てられているものとする必要がある。

【 0 0 1 6 】

そこで、本発明においては、ストレージ 1 は各計算機 2 x ごとにその計算機 2 x が使用する内部 LU を、計算機から見たときに 0 から始まる連続した論理ボリューム番号になるよう再定義することで上記の課題を解決する。このように各計算機 2 x から認識される LU を外部論理ボリューム（外部 LU）、その番号を外部論理ボリューム番号（外部 LUN）と呼び、内部 LU および内部 LUN と区別する。本発明では外部 LU と内部 LU の間に両者の関連を定義する LUN 結合情報を設け、この LUN 結合情報によって外部 LU と内部 LU との結合の構成変更が可能となる。これらの外部 LUN、LUN 結合情報および内部 LUN の対応関係は、ストレージ 1 が備える LUN マップ 1 1 1 で管理する。

【 0 0 1 7 】

ストレージ 1 が備える LUN マップ 1 1 1 の一例を図 2 に示す。LUN マップ 1 1 1 には、ポート番号、Target ID、外部 LUN、LUN 結合情報、内部 LUN、WWN、S_ID および属性を格納する。以下、これらについて説明する。

【 0 0 1 8 】

ポート番号は、ストレージ 1 が備えるファイバチャネル接続ポートの番号を格納する。本実施例ではポート数は 1 個を仮定し、一律 0 を格納する。

【 0 0 1 9 】

Target ID は、計算機 2 x とストレージ 1 との接続インターフェイスにおいて、ストレージ 1 に割り当てる識別 ID のことである。本実施例のように計算機 2 x とストレージ 1 との接続インターフェイスがファイバチャネルの場合には各ポート毎に唯一の D_ID (Destination ID) を備えるが、ポート番号の項があるので省略してもよいし、ファイバチャネルの接続ポートの初期化時に決定した D_ID を格納しても良い。SCSI の場合には同一ポートに複数の ID を備えることができるので、そのときの各 LUN の属する Target ID を格納する。本実施例ではファイバチャネルを仮定しているので、Target

I D の欄は未使用とし、一律 0 を格納する。

【 0 0 2 0 】

外部 L U N、L U N 結合情報および内部 L U N とは、それぞれの L U N の対応関係を表す。まず、ストレージ 1 の物理ディスクは内部 L U N が 0 から $n - 1$ とされた n 個の論理ボリュームの領域と、内部 L U N が $k + 0$ から $k + m$ とされた m 個の論理ボリュームの領域となるものとされている。前者の領域に対しては外部 L U N としては 0 が割り当てられており、後者の領域に対しては外部 L U N としては 1 が割り当てられている。L U N 結合情報は外部 L U N と内部 L U N との結合関係を示す情報であり、左側に外部 L U N、右側に内部 L U N を表示し、両者をハイフンで結んで結合関係を示している。

【 0 0 2 1 】

先頭 L B A は、各内部 L U の先頭アドレスが、計算機 2 x からみた場合、外部 L U 内のどのアドレスに割り当てられているかを示す。L B A（論理ブロックアドレス）とは、L U 内のアドレスのことであり、計算機 2 x はこのアドレスを用いて L U 内のデータにアクセスする。ある外部 L U が 1 つの内部 L U から構成されている場合には、先頭 L B A には 0 が割り当てられる。L U の構成の変更により複数の内部 L U から構成される場合は、先頭 L B A が書き換えられるが、この点については、L U の構成の変更例について説明するときに具体的に説明する。

【 0 0 2 2 】

ブロック数は、各内部 L U の論理ブロック数を表し、これにより各内部 L U のサイズを知ることができる。

【 0 0 2 3 】

WWN は、各計算機 2 x を特定する情報である World Wide Name を格納する。ファイバチャネルの接続ポートとポートの接続関係を確立するポートログイン処理の際に、各計算機 2 x の WWN がストレージ 1 に通知される。

【 0 0 2 4 】

S _ I D は、ファイバチャネルのフレームヘッダに格納される I D 情報であり、フレームを作成したソース（イニシエータ）を識別する I D である。S _ I D は、ファイバチャネルの初期化の際に、動的に割り当てられる。先に述べた WW

Nは初期化の際に交換された各ファイバチャネルの接続ポートにより一意に設定される値であるが、WWNとS__IDの関連づけを行うことで、フレーム毎にWWNを調べなくてもS__IDのみ検査することで計算機2xが特定できるようになっている。

【0025】

属性は、各LUの所有属性を示す。「専用」は1台の計算機2x専用のLUであることを示す。「共用」は複数の計算機2xが共用するLUであることを示す。

【0026】

図2に示すLUNマップ111を見ると、計算機2aから2nのそれぞれに内部LUNが0からn-1のそれぞれの専用の内部LUが割り当てられており、これらのブロック数は1048576であることがわかる。また、これらの内部LUNは連続番号とされていても、外部LUNはすべて0とされている。さらに、共用のLUに対して、k+0からk+mの連続番号とされた内部LUNの領域が設定されており、これらのブロック数は4194304であることがわかる。これらの領域の内部LUに対しては、外部LUNはすべて1とされている。したがって、各計算機はOSブート時にストレージをサーチする際、まず、外部LU0を、次いで1をサーチことにより自分が使用できる内部LUを知ることができる。

【0027】

次に論理ボリュームLUの構成変更について図3のフローチャートを用いながら説明する。

【0028】

管理コンソール4を操作する作業者は、管理ユーティリティ41からLUの構成変更指示を送る。LUを拡張する場合、拡張したい内部LUNと、新たに結合する内部LUNを指定する。この指示は、通信手段7によりストレージ1に発行される(ステップ701)。

【0029】

ストレージ1のLU制御手段11はこの指示を受け取り、指定されたLUが正

しいか否かを判定する（ステップ702）。存在しないLUNが指定されていたり、新たに結合しようとする内部LUが、別の計算機2xに割り当てられていたりする等、指定された内部LUが正しくない場合、LU制御手段11は管理ユーティリティ41にエラーを返し、終了する（ステップ708）。指定されたLUが正しい場合、LU制御手段11は、その指示に従いLUNマップ111を書き換える（ステップ703）。

【0030】

次に、計算機2xのLU認識手段211xにLUの構成が変更されたことを通知する。この通知方法にはLU制御手段11がファイバチャネル5x経由でLU認識手段211xに送信する方法と、ストレージ1のLU制御手段11が管理ユーティリティ41に通知し、管理ユーティリティ41からLAN6x経由で計算機2xのLU認識手段211xに通知する方法と、管理者が直接計算機2xを操作し、LU認識手段211xに通知する方法があり、任意の方法が使用できる（ステップ704）。

【0031】

LU認識手段211xは、LU制御手段11から変更後のLUサイズを取得する。LU認識手段211xは、構成変更の通知を受けずにLU制御手段11に構成情報の提示を要求することもある。その場合、構成変更の通知（ステップ704）は省略できる（ステップ705）

LU認識手段211xは変更されたLUNとそのサイズを論理ボリューム管理ソフト21x（FS又はLVM）に通知し、計算機2x上で変更後のLUが使用可能となる（ステップ706）。

【0032】

このように、既存LUのオンラインアクセスを継続しながら、構成の変更を反映させることができる。

【0033】

次に、LU構成の変更例を具体的に示し、LUNマップ111の書き換えについて詳しく説明する。LU構成変更の例として、計算機2aのLU0領域を拡張する場合を考える。図4は物理ディスク領域12の内部LUの状態を示す。12

0 x (1 2 0 a - 1 2 0 c) は内部 L U である。物理ディスク領域 1 2 内に L U 0 領域 1 2 0 a、L U 1 領域 1 2 0 b、L U 2 領域 1 2 0 c が連続的に構成されている。今、説明を簡単にするために、L U 0 領域 1 2 0 a が計算機 2 a に割り当てられ、L U 1 領域 1 2 0 b が計算機 2 b に割り当てられていて、L U 2 領域 1 2 0 c は、どの計算機にも割り当てられていない内部 L U であるものとする。この例では、L U 2 領域 1 2 0 c がどの計算機にも割り当てられていない内部 L U であっても、L U 0 領域 1 2 0 a の次に連続的に L U 1 領域 1 2 0 b が形成されているため、L U 0 領域 1 2 0 a 領域を物理的に拡張することはできない。この状態では、計算機 2 a には L U 0 領域 1 2 0 a が自己の論理ボリュームとして見えおり、計算機 2 b には L U 1 領域 1 2 0 b が自己の論理ボリュームとして見えている。

【 0 0 3 4 】

図 5 はこのときの L U N マップ 1 1 1 を示す。計算機 2 a (W W N a) に内部 L U N 0 領域が、計算機 2 b (W W N b) に内部 L U N 1 領域がそれぞれ割り当てられ、内部 L U N 2 領域はフリー（各項目は未定義）である。ここで、計算機 2 a の L U の拡張要求があると、図 3 で説明したように、L U 制御手段 1 1 は、管理ユーティリティ 4 1 からの構成変更指示に基づき、内部 L U 2 1 2 0 c を計算機 2 a の内部 L U とするように L U N マップ 1 1 1 を書き換える。図 6 (a) はこのようにして書き換えられた L U N マップ 1 1 1 を示す。これから分かるように、計算機 2 a は内部 L U N が 0 と 2 とを自己が使用できる L U として認識できる。そして、内部 L U N 2 は内部 L U N 0 に連続して使用されるものとなるから、内部 L U N 2 の先頭 L B A のブロック数から始まるものとされる。この処理は L U 制御手段 1 1 で行われる。また、この処理に対応して、W W N、S _ I D 等も計算機 2 a に対応するものを格納する。計算機 2 a から L U にアクセスする場合、先頭 L B A によって内部 L U 0 と内部 L U 2 のどちらへのアクセスかわかる。

【 0 0 3 5 】

図 7 は、このように書き換えられた物理ディスク領域 1 2 の内部 L U の状態を説明する図である。図 4 と対照して明らかなように、内部 L U 1 2 0 x (1 2 0

a-120c) が連続して形成されていることに変わりはないが、一点鎖線で囲って示す領域122は、計算機2aのLUとして機能するものとなる。この状態では、計算機2aにはLU0領域120aとLU2領域120cとを合わせた領域122が自己の使用できる一つの論理ボリュームとして見えており、計算機2bにはLU1領域120bが自己の論理ボリュームとして見えている。

【0036】

図6(b)、(c)は、LU構成の変更後の他の例についてのLUNマップ111を表している。図6(b)は、上述した計算機2aへの内部LU0に内部LU2を追加した論理ボリュームの追加に加えて、さらに内部LU5を追加した状態を示す。一方、図6(c)は、計算機2aへの内部LU0に内部LU5を追加した論理ボリュームの追加に加えて、さらに内部LU2を追加した状態を示す。すなわち、図6(b)、(c)では、内部LUNの追加の順番が異なる例を示す。この順番が異なることは、計算機2aから見て、別に意味があるわけではなく、LU制御手段11が先頭LBAおよび構成変更に伴うWWN、S__ID等の情報を適正に格納してさえいれば、図6(b)、(c)のいずれの場合でも、単位ブロック数の3倍のブロック数を持つ論理ユニットを自己の計算機の機能の実行のために使用できる。

【0037】

以上のように仮想的にLU領域の管理を行なうことによって、物理的には不可能なLUの拡張等の構成変更が可能となる。

【0038】

本実施例によれば、ストレージ装置内部の論理ボリューム領域の構成を自由に変更することができるという効果がある。また、本実施例によれば、既存論理ボリュームへのアクセスを継続させながら論理ボリューム領域を拡張することができるという効果がある。さらに、本実施例によれば、ストレージ装置内部で構成の変更を行なえるので、計算機のOSに非依存で論理ボリューム領域を拡張することができるという効果がある。

【0039】

(実施例2)

図 8 は本実施例の計算機システムの構成図である。計算機システムの構成における実施例 1 との相違点は、ストレージ 1 にコピー手段 1 3 を追加したことだけであり、その他は同一である。コピー手段 1 3 は、ある LU を別の領域にコピーする手段である。

【 0 0 4 0 】

LU の構成変更について図 9 のフローチャートを用いて説明する。管理コンソール 4 を操作する作業者は、管理ユーティリティ 4 1 から LU の構成変更指示を送る。LU を拡張する場合、拡張したい内部 LU N と、拡張するサイズを指定する。この指示は、通信手段 6 によりストレージ 1 に発行される（ステップ 8 0 1）。

【 0 0 4 1 】

ストレージ 1 の LU 制御手段 1 1 はこの指示を受け取り、指定された内部 LU のサイズと拡張分のサイズをあわせた合計サイズが空き領域から確保可能かどうかを判定する（ステップ 8 0 2）。

【 0 0 4 2 】

領域が確保できない場合、LU 制御手段 1 1 は管理ユーティリティ 4 1 にエラーを通知し、終了する（ステップ 8 1 1）。領域確保が可能ならば LU 制御手段 1 1 はその領域を確保し、LU 認識手段 2 1 1 x に新領域のアドレスを通知する（ステップ 8 0 3）。LU 認識手段 2 1 1 x は新領域のアドレスを FS/VLM 2 1 に通知する（ステップ 8 0 4）。この二つのステップは本実施例によるコピー中に計算機が当該論理ユニット LU にアクセスしたときのデータ不一致を避けるためであるが、この点については後述する。

【 0 0 4 3 】

次いで、LU 制御手段 1 1 はコピー手段 1 3 にコピー指示を送る（ステップ 8 0 5）。コピー指示を受けたコピー手段 1 3 は指定された LU のコピーを行なう。LU のコピーが終了すると、コピー手段 1 3 は LU 制御手段 1 1 にコピー終了を通知する（ステップ 8 0 6）。

【 0 0 4 4 】

コピー終了の通知を受けると LU 制御手段 1 1 は LU N マップ 1 1 1 を書き換

え、新しくできたLUを計算機2xに割り当てる（ステップ807）。

【0045】

ステップ807以降の処理は実施例1と同様である。

【0046】

LU構成の変更を具体的に説明する。図10は本実施例の構成変更例を説明する図である。図10（a）は変更前の内部LUの状態を示す。図4で説明したのと同様に、計算機2aのLU0領域を拡張する場合を考える。物理ディスク領域12内にLU0領域120a、LU1領域120bが連続的に構成され、LU0領域120aが計算機2aに割り当てられ、LU1領域120bが計算機2bに割り当てられている。この例では、これらの領域以外は空き領域とされているものとする。この例でも、十分な空き領域があったとしても、LU0領域120aの次に連続的にLU1領域120bが形成されているため、LU0領域120a領域を物理的に拡張することはできない。このとき、計算機2aにはLU0領域120aが自己の論理ボリュームとして見えており、計算機2bにはLU1領域120bが自己の論理ボリュームとして見えていることは、第1の実施例の場合と同じである。

【0047】

LUの拡張に関するコピーについてまず説明する。管理コンソール4を操作する作業者から、管理ユーティリティ41を通して拡張したい内部LUNと、拡張するサイズを指定してLUの構成変更指示がLU制御手段11に送られると、LU制御手段11は、空き領域123から拡張したい内部LUNの領域サイズと拡張分の領域サイズとの大きさの新らしいLU2の領域を内部LUNとして確保する。領域が確保できない場合、エラーとして、管理ユーティリティに通知する。次にLU制御手段11は、確保した領域LU2 120dの先頭位置からLU0 120aの内容をコピーするようコピー手段13に指示する。図10（b）はコピー手段13によるコピーの状態を説明する図である。領域LU2 120dに破線で示すように、LU0 120aからLU2 120dにコピーが行なわれる。LU0 120aの内容を全てLU2 120dにコピーし終わると、コピー手段13はLU制御手段11にコピー終了を通知する。図10（c）はコピ

ー完了後の内部LUの状態である。コピー終了通知を受けたLU制御手段11は、LUNマップ111を書き換え、LU2 120dを計算機2aに割り当てる。LU0 120aは未定義とされる。

【0048】

図11は更新前のLUNマップ、図12は更新後のLUNマップを表す。更新前LUNマップでは、内部LUN 0に計算機2aが割り当てられていたが、更新後LUNマップでは内部LUN 2に計算機2aが割り当てられていることがわかる。しかも、この内部LUN 2のブロック数は当初の2倍のブロック数とされている。すなわち領域の拡大が新しい領域により実現されたことが分かる。このようにコピー手段を用いることにより、元のLU内のデータをそのまま継続しながら新内部LUを作成し、LU拡張等の構成変更が可能となる。

【0049】

ところで、LUの領域変更のためのコピー中に計算機からLUへアクセスする場合がある。この場合、このアクセスが読み出しである場合には、元のLU領域から読み出せば良い。もし、書き込みのアクセスであった場合には、該当するアドレスの元のLU領域と新しいLU領域の両方にアクセスして書き込みをさせることとしないと、データの不一致が起こる可能性がある。前述したステップ803、804がこれを防止するためのものである。すなわち、この書き込みのアクセスが、コピー手段13がまだコピーを完了していない元のLU領域であった場合には問題が無い。しかし、すでにコピーを完了している元のLU領域であった場合には、そのアドレスのデータを更新しても、この新しいデータが読み出されて新しいLU領域にコピーされることは無いから、ステップ806でコピー終了が通知されるとデータの不一致が起こる。このようなデータの不一致が起こる可能性がある場合には、新しいLU領域のアドレスを計算機に教えてデータの書き込みは両方のアドレスに対して行うようにすることが必要である。もっとも、計算機は従来の通り通常アドレスに書き込むだけとしておいてLU制御手段11がステップ802からステップ807までの間は両方のアドレスに書き込むようにしても良い。

【0050】

以上のように、新たに確保した領域に既存LUの内容をコピーすることで、既存データを活用しながらLUの領域拡張等を行なうことができる。また、1つの外部LUに対して、1つの内部LUが割り当てられるので、管理が容易になる。本実施例によれば、上記実施例1と同一の効果があるとともに、論理ボリューム領域が物理的な連続領域に割り当てられるので、論理ボリュームの管理が容易になるという効果がある。

(実施例3)

実施例3を説明する。本実施例の計算機システムは実施例2と同一である。本実施例は上記実施例1における論理ボリュームの仮想的な結合と、実施例2のコピー手段による拡張の組み合わせで実現される。

【0051】

LUの構成変更について図13のフローチャートを用いて説明する。

【0052】

管理コンソール4を操作する作業者は、管理ユーティリティ41からLUの構成変更指示を送る。LUを拡張する場合、拡張したい外部LUに割り当てられている全ての内部LUNと、拡張するサイズを指定する。この指示は、通信手段6によりストレージ1に発行される（ステップ901）。

【0053】

ストレージ1のLU制御手段11はこの指示を受け取り、指定された全ての内部LUのサイズと拡張分のサイズをあわせた合計サイズが空き領域から確保可能かどうかを判定する（ステップ902）。

【0054】

領域確保可能であれば、コピー手段を用いて指定された全てのLUをコピーし、新内部LUを作成する。なお、この段階でも、実施例2で説明したデータの不一致を回避するための処置は取られる。以降の処理は実施例2と同様である。（ステップ903）

全ての合計サイズが空き領域から確保できない場合、LU制御手段11は仮想的な結合が可能かどうかを判定する。拡張サイズ分の領域が確保できれば仮想的な結合が可能である（ステップ904）。

【 0 0 5 5 】

仮想的な結合が可能な場合、確保した拡張サイズ分の領域を、拡張したいLUに仮想的に結合する（ステップ905）。以降の処理は実施例1と同様である。

【 0 0 5 6 】

仮想的な結合が不可能であれば、LU制御手段11は管理ユーティリティ41にエラーを返し、終了する（ステップ906）。

【 0 0 5 7 】

実施例3によるLU構成の変更結果の例を図14を用いて説明する。12は物理ディスク領域、121x（121e－121g）は仮想内部LUである。図14（a）は実施例2と同様に仮想内部LU0 121eを拡張しようとしたが領域が不足であったために、新内部LU2 121gが構成されるとともに、図7と同様に、実施例1による論理ボリュームの仮想的な結合が行われて新内部122が構成された状態を示す。図14（b）は実施例2と同様に拡張すべき領域を持つ新たな新内部LU2 121gが構成されるとともに、内部LU0が開放された状態を示す。いずれの場合でも、LUNマップ111はそれぞれに対応した内容に更新される。

【 0 0 5 8 】

以上のように実施例1における論理ボリュームの仮想的な結合と、実施例2のコピー手段による拡張を組み合わせることによって、より柔軟な論理ボリュームの構成変更が可能となる。

【 0 0 5 9 】

なお、実施例の説明からも理解できるように、本発明の論理ボリューム番号マップ111では、実施例のように、ポート番号、Target ID、外部LUN、LUN結合情報、内部LUN、先頭LBA、ブロック数、WWN、S_IDおよび属性の全てを備えるものである必要は無い。たとえば、ブロック数自体は内部LUをどれだけの大きさのものを単位として扱うものとするかを定義しておけば良いのであって、これが、論理ボリューム番号マップ111の要素となっている必然性は無い。同様に、先頭LBAも、図6を参照して分かるように、分離された複数の内部LUを一つの外部LUに対応させているときの直前までの内部

LUのブロック数の累積として計算できるものであるから、必要の都度計算するものとすれば良い。さらに、S__IDおよび属性もあったほうが、管理者に分かりやすいと言う以上の意味は無い。要は計算機から見える外部LUNとストレージの内部LUNとの結合が明確に定義されており、この組み合わせが、どの計算機のためのものであるかが分かるものであれば良い。

【 0 0 6 0 】

【発明の効果】

本発明によれば、ストレージ装置内部の論理ボリューム領域の構成を自由に変更することができるという効果がある。さらに、既存論理ボリュームへのアクセスを継続させながら論理ボリューム領域を拡張することができるという効果がある。さらに、ストレージ装置内部で構成の変更を行なえるので、計算機のOSに非依存で論理ボリューム領域を拡張することができるという効果がある。

【図面の簡単な説明】

【図 1】

実施例 1 の計算機システムの構成図。

【図 2】

実施例 1 のストレージ 1 が備える LUN マップの一例を示す図。

【図 3】

実施例 1 の論理ボリューム LU の構成変更のフローチャートを示す図。

【図 4】

実施例 1 の物理ディスク領域の内部 LU の状態を示す図。

【図 5】

実施例 1 の内部 LU の変更前の LUN マップの例を示す図。

【図 6】

(a) - (c) は実施例 1 の内部 LU の変更後の LUN マップの例を示す図。

【図 7】

図 6 (a) の物理ディスク領域の内部 LU の状態を示す図。

【図 8】

実施例 2 の計算機システムの構成図。

【図 9】

実施例 2 の論理ボリューム LU の構成変更のフローチャートを示す図。

【図 1 0】

(a) - (c) は実施例 2 による論理ボリューム LU の構成変更に対応した物理ディスク領域の内部 LU の状態を示す図。

【図 1 1】

実施例 2 の内部 LU の変更前の LUN マップの例を示す図。

【図 1 2】

実施例 2 の内部 LU の変更後の LUN マップの例を示す図。

【図 1 3】

実施例 3 の論理ボリューム LU の構成変更のフローチャートを示す図。

【図 1 4】

(a)、(b) は実施例 3 による論理ボリューム LU の構成変更に対応した物理ディスク領域の内部 LU の状態を示す図。

【図 1 5】

従来技術のストレージの構成例を示す図。

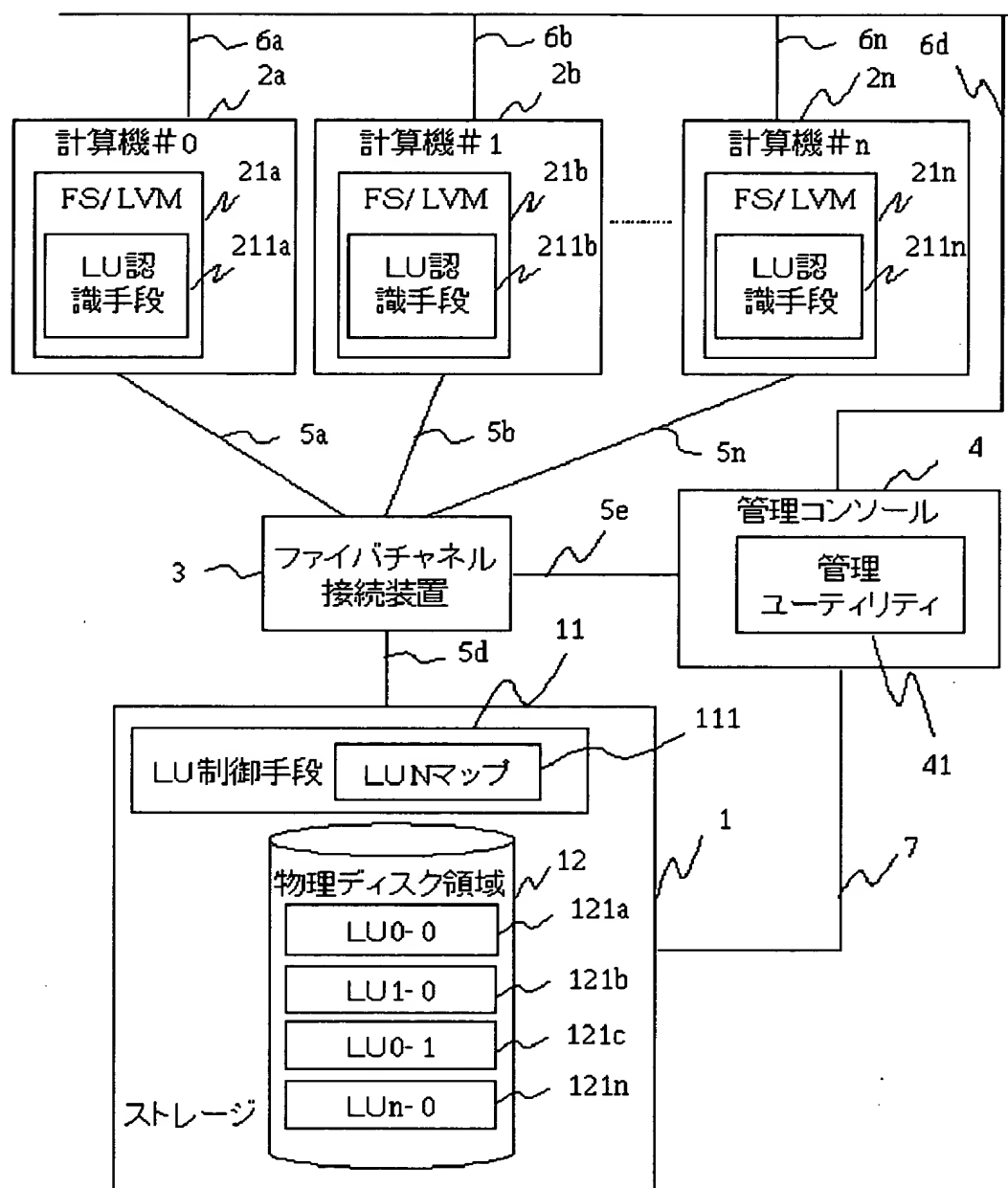
【符号の説明】

1 : ストレージ、1 1 : LU 制御手段、1 1 1 : LUN マップ、1 2 : 物理ディスク領域、1 2 0 : 内部 LU、1 2 1 : 内部 LU、1 2 2 : 外部 LU、1 2 3 : 空き領域、1 3 : コピー手段、2 : 計算機、2 1 : FS / LVM、2 1 1 : LU 認識手段、3 : ファイバチャネル接続装置、4 : 管理コンソール、4 1 : 管理ユーティリティ、5 : ファイバチャネル、6 : LAN、7 : 通信インターフェイス、1 0 0 : ストレージ、1 0 1 : 物理ディスク領域、1 0 1 1 : 内部 LU、1 0 2 : 追加ディスク領域。

【書類名】 図面

【図 1】

図 1



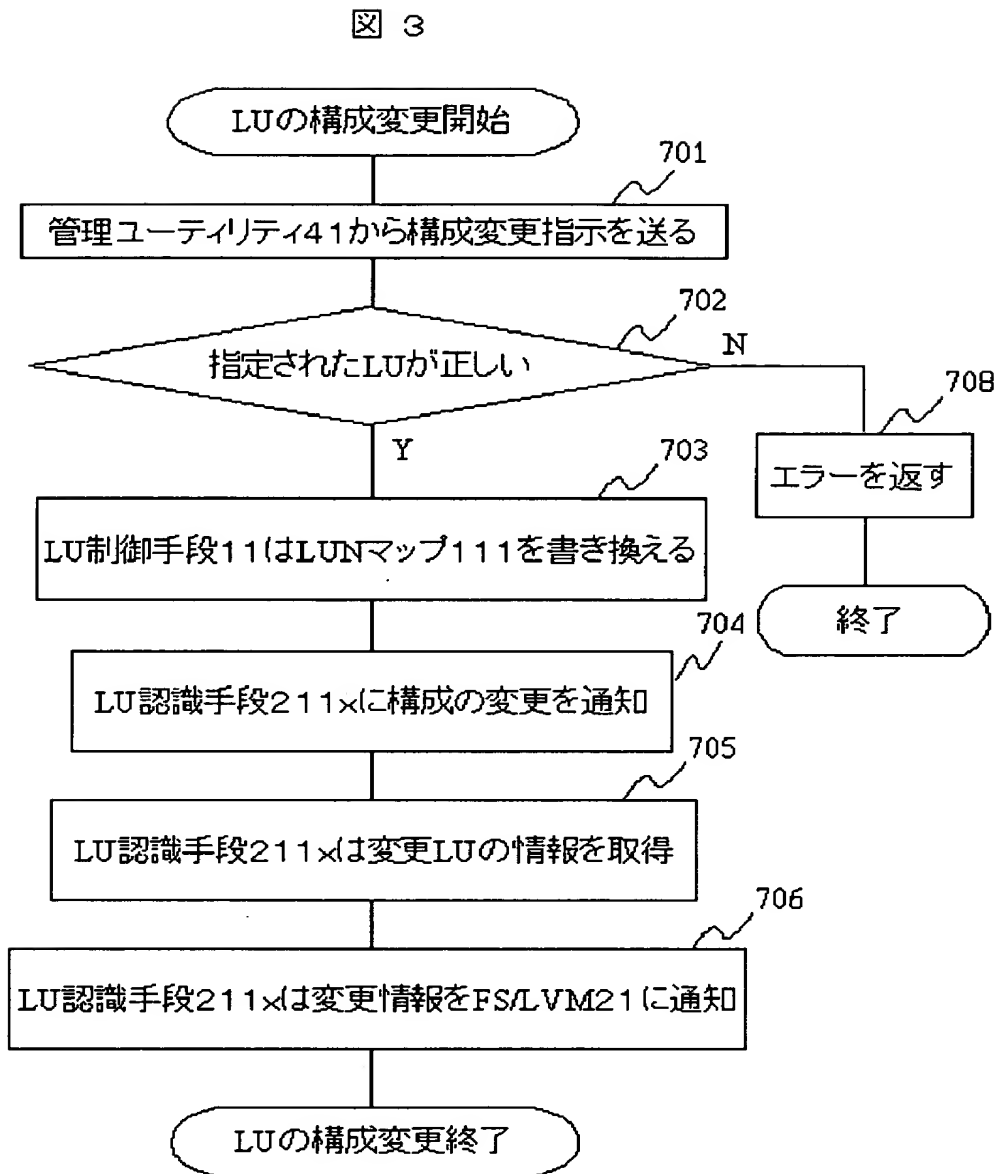
【図 2】

図 2

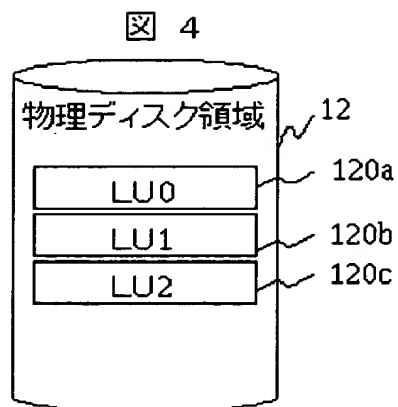
111

ポート 番号	Target ID	外部 LUN	LUN結 合情報	内部 LUN	先頭 LBA	ブロッ ク数	WWN	S_ID	属性
0	0	0	0-0	0	0	1048576	WWNa	S_IDa	専用
0	0	0	0-1	1	0	1048576	WWNb	S_IDb	専用
0	0	0	0-2	2	0	1048576	WWNc	S_IDc	専用
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	0	0	0-(n-1)	n-1	0	1048576	WWNn	S_IDn	専用
0	0	1	1-(k+0)	k+0	0	4194304	WWNa	S_IDa	共用
0	0	1	1-(k+1)	k+1	0	4194304	WWNc	S_IDc	共用
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	0	1	1-(k+m)	k+m	0	4194304	—	—	共用

【図 3】



【図 4】



【図 5】

図 5

ポート 番号	Target ID	外部 LUN	LUN結 合情報	内部 LUN	先頭 LBA	ブロッ ク数	WWN	S_ID	属性
0	0	0	0-0	0	0	1048576	WWNa	S_IDa	専用
0	0	0	0-1	1	0	1048576	WWNb	S_IDb	専用
-	-	-	-	2	-	-	-	-	-
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

111

【図 6】

(a) 図 6 111

ポート 番号	Target ID	外部 LUN	LUN結 合情報	内部 LUN	先頭 LBA	ブロッ ク数	WWN	S_ID	属性
0	0	0	0-0	0	0	1048576	WWNa	S_IDa	専用
0	0	0	0-1	1	0	1048576	WWNb	S_IDb	専用
0	0	0	0-2	2	1048576	1048576	WWNa	S_IDa	専用
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

(b) 111

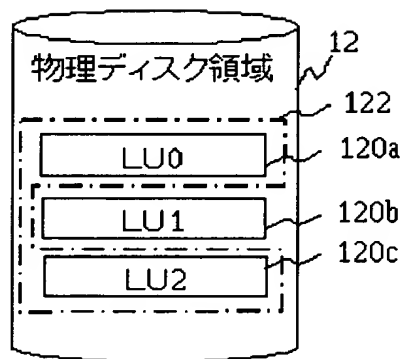
ポート 番号	Target ID	外部 LUN	LUN結 合情報	内部 LUN	先頭 LBA	ブロッ ク数	WWN	S_ID	属性
0	0	0	0-0	0	0	1048576	WWNa	S_IDa	専用
0	0	0	0-1	1	0	1048576	WWNb	S_IDb	専用
0	0	0	0-2	2	1048576	1048576	WWNa	S_IDa	専用
0	0	0	0-5	5	2097152	1048576	WWNa	S_IDa	専用
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

(c) 111

ポート 番号	Target ID	外部 LUN	LUN結 合情報	内部 LUN	先頭 LBA	ブロッ ク数	WWN	S_ID	属性
0	0	0	0-0	0	0	1048576	WWNa	S_IDa	専用
0	0	0	0-1	1	0	1048576	WWNb	S_IDb	専用
0	0	0	0-5	5	1048576	1048576	WWNa	S_IDa	専用
0	0	0	0-2	2	2097152	1048576	WWNa	S_IDa	専用
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

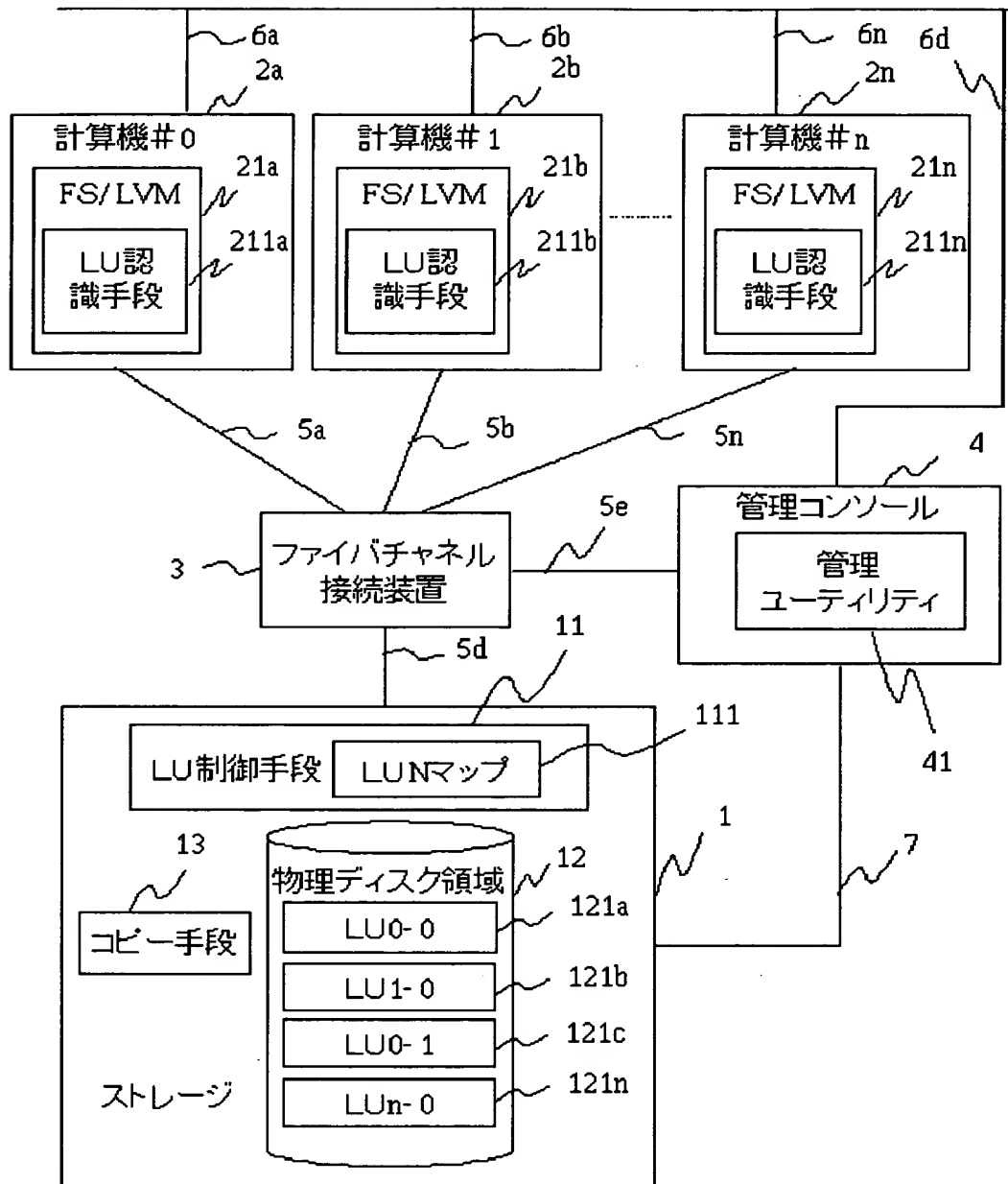
【図 7】

図 7

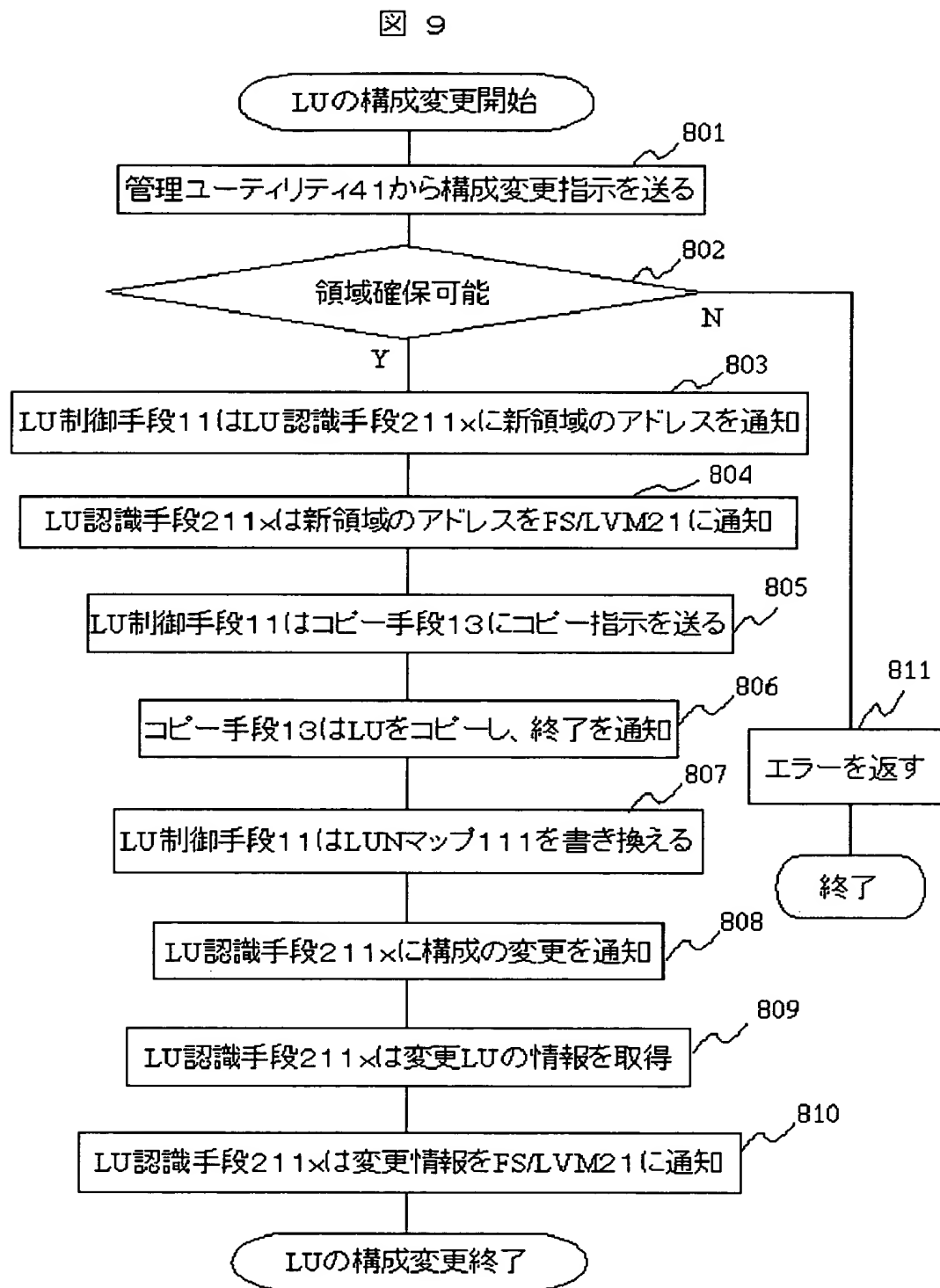


【図 8】

図 8

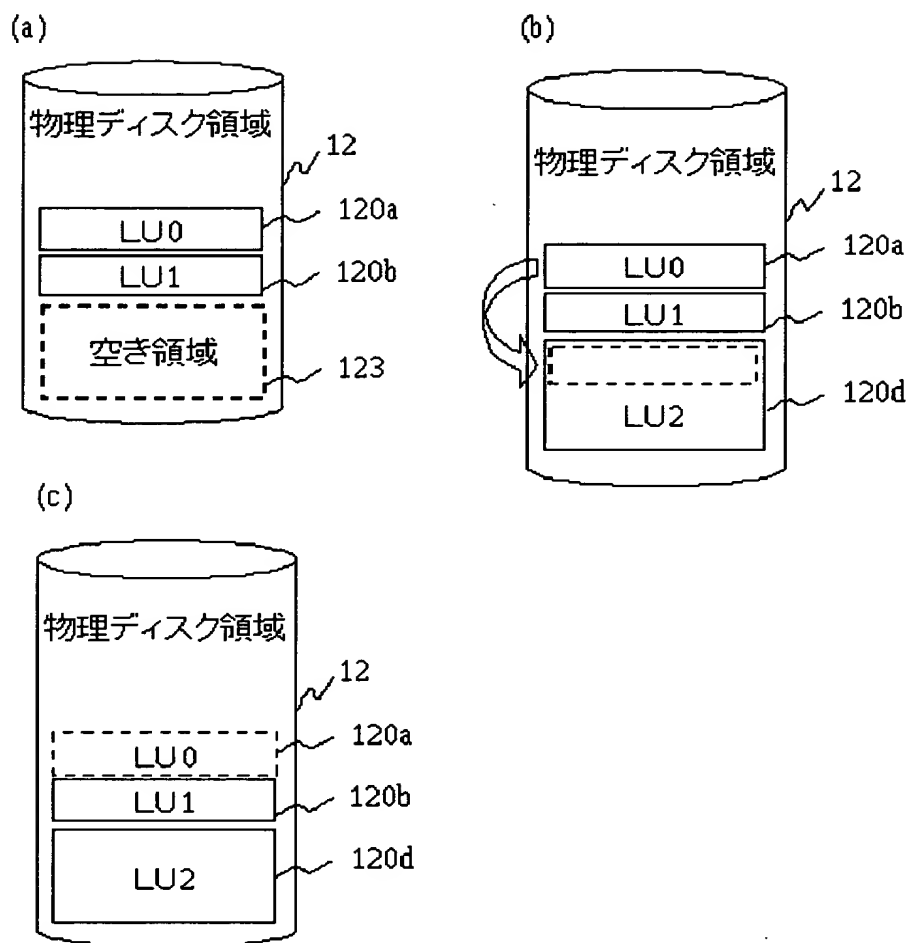


【図 9】



【図 1 0】

図 10



【図 1 1】

図 11

111

ポート 番号	Target ID	外部 LUN	LUN結 合情報	内部 LUN	先頭 LBA	ブロッ ク数	WWN	S_ID	属性
0	0	0	0-0	0	0	1048576	WWNa	S_IDa	専用
0	0	0	0-1	1	0	1048576	WWNb	S_IDb	専用
-	-	-	-	2	-	-	-	-	-
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

【図 1 2】

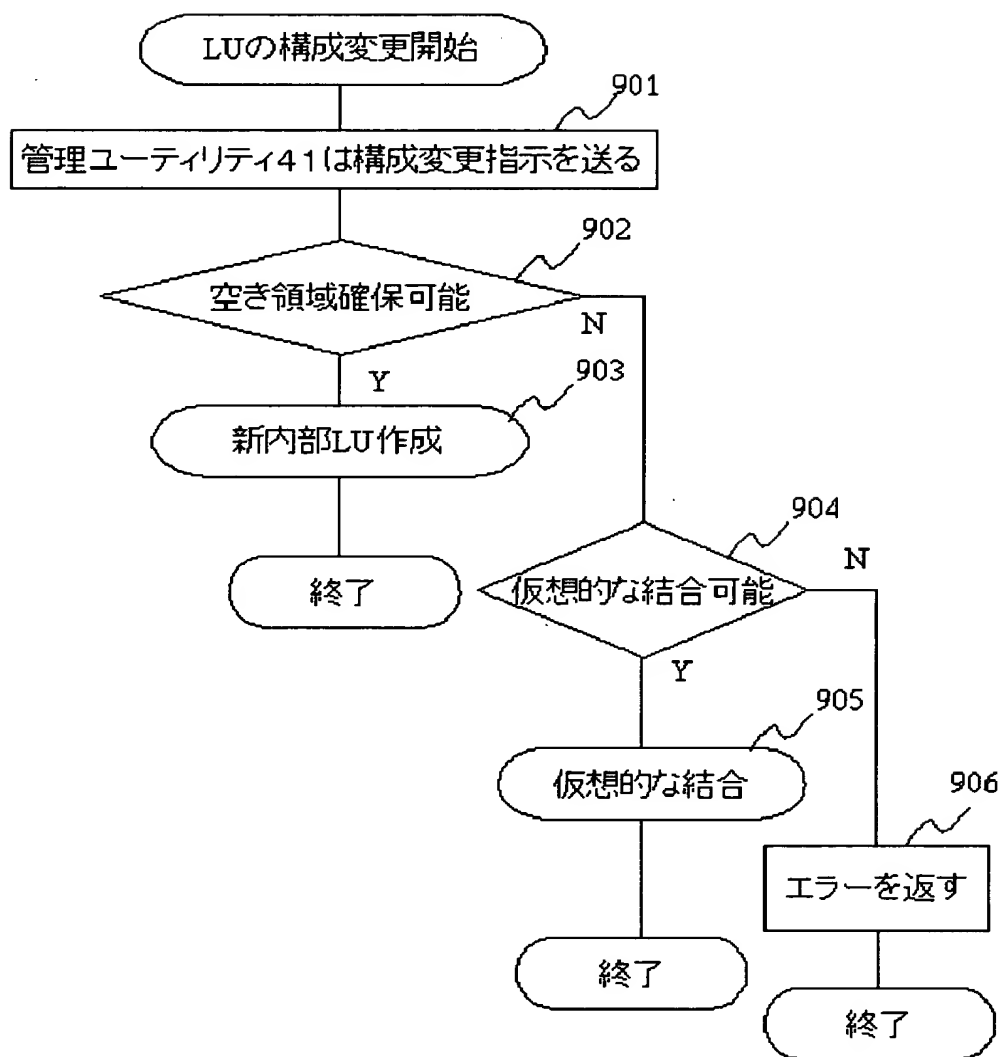
図 12

111

ポート 番号	Target ID	外部 LUN	LUN結 合情報	内部 LUN	先頭 LBA	ブロッ ク数	WWN	S_ID	属性
-	-	-	-	0	-	-	-	-	-
0	0	0	0-1	1	0	1048576	WWNa	S_IDb	専用
0	0	0	0-2	2	0	2097152	WWNa	S_IDa	専用
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

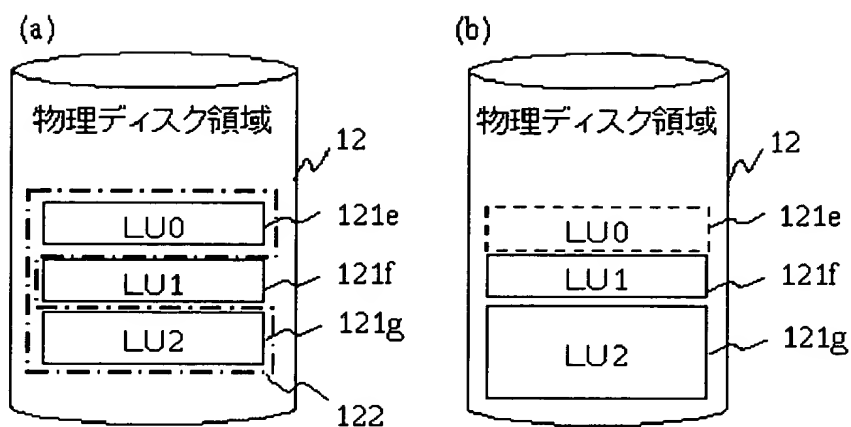
【図 1 3】

図 13



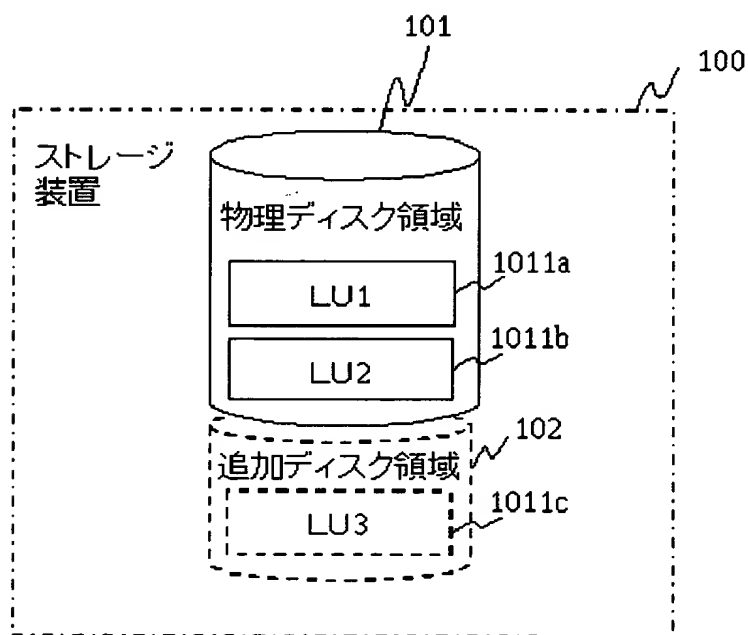
【図 1 4】

図 14



【図 1 5】

図 15



【書類名】 要約書

【要約】

【課題】 継続して論理ボリュームを使用しながら論理ボリューム個別の領域拡張を可能とする。

【解決手段】 ストレージは論理ボリュームの構成制御を行なう論理ボリューム制御手段および論理ボリュームの構成情報を記述する論理ボリューム番号マップを備える。論理ボリューム番号マップに一つの外部論理番号に二つ以上の内部論理番号の記述を許すことにより、ストレージ内部の論理ボリュームの結合の自由度を上げる。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 5 1 0 8]

1. 変更年月日	1 9 9 0 年 8 月 3 1 日
[変更理由]	新規登録
住 所	東京都千代田区神田駿河台 4 丁目 6 番地
氏 名	株式会社日立製作所